Haaga-Helia
University of Applied Sciences

Customer Information Management System

Case Company: *Punjab Building Products Ltd.*

Bachelor's Thesis
Degree Programme BITe
2015

Haaga-Helia
University of Applied Sciences

Abstract

Date: 10.11.2015

Author(s)

Abdul Rehman Khan Ranizai

Degree programme

Business Information Technology (BITe)

Report/thesis title	**Number of pages and appendix pages**
Customer Information Management System Case Company: *Punjab Building Products Ltd.*	52

Thesis is based on developing a Customer information management system for (PBP) Punjab building products ltd. The PBP ltd. is a public limited company and it started its activities in 1980 and involved in the manufacturing of building products. PBP Ltd. was struggling to find a secure method of saving company's customers data and information because most of the work done on Microsoft excel sheets or word document, which was a big risk for the company information data security.

The main task and objective of the project is to design and build a prototype of the customer information management system for PBP (Punjab Building Products Ltd). Since, CIM system is not used at PBP ltd., I have managed to give a complete computerized I.T support management system wherein all the functions and activities of information related to customer, product, supplier and delivery is controlled, monitored and readily available to all concerned in the company.

The theoretical part of the thesis include company's information, project scope, stakeholders, system architecture, system requirement specification, non-functional requirements and functional requirements. Whereas the empirical part of the thesis is dedicated for defining use case for system requirements, usability requirements, designing and implementing the desired product. Moreover, detail description of techniques used in this project are included in the technical knowledge.

The implementation of the final version of the product and deployment of the new system are not included in the project because the application size is very large and the time period available for implementation is not enough to implement the complete web application during thesis project. I will continue to work on the system implementation and will finalize the application for company use which is aimed to be released in January 2017 with some more advanced features.

The thesis provides complete guidelines for all those developers interested in developing customer information management system for real world companies.

Keywords

PBP, CIM, ASP.NET MVC, System requirements, System architecture, System administrator, Central database, Data CRUD matrix, Turnkey jobs, Interface metaphor, Usability requirements, Prototype.

Table of Contents

1 Introduction

1.1 The background of the project

PBP (Punjab Building Products Ltd.) is a public limited company incorporated in Pakistan. Since 1980, it is engaged in the manufacturing and sales of building products, construction materials, asbestos cement pressure pipes, Comete joints, rubber rings, uPVC (un-plasticized polyvinylchloride) plastic pressure piping systems, uPVC fittings and lot more. From the production of the very first pipe PBP set its standard high and thus established its benchmark making the customers more confident. PBP has the capacity and capability to make all its products which not only comply with relevant national, international and British standards, but also meet its own strict criteria of consistent high quality and performance. To ensure these high standards, PBP follows a policy of using best quality of imported raw material formulations. These, in combination with latest automated controlled equipment, produce a high quality and guaranteed end-product for the customers use.

Moreover, **PBP Ltd**. has also introduced a construction management division / CMD. This CMD division not only caters to in house construction needs but also takes up turnkey jobs for those customers who wish to get the Punjab building products installed by the engineers of PBP Ltd. This CMD division also guides and helps those customers who wish to execute their projects themselves, by providing guidelines on all possible unforeseen difficulties and briefing them the solutions in advance thus minimizing the overall costs to the customers. The CMD division regularly conducts workshop sessions for the customer's design engineering staff to help them understand the proper laying and jointing of pressure piping systems, proper positioning of the air-valves, sluice-valves and check-valves in their pipeline projects. The CMD division also assists them in carrying out the pressure testing of the laid pipelines all over the country as per the standard testing procedures. Thus the PBP ltd., offers pre-sales, during - sales and after-sales services for all its customers throughout the country.

All this is done because the company is very rapidly changing for the best, and adding and diversifying their shelves to suit the customer's needs. Construction management division specializes in the execution and management of all types of civil engineering projects, and their highly skilled technical and motivated staff has amongst themselves a total of more than 25 years of cumulative construction Industry experience.

The Mission of Punjab Building Products Ltd. includes:

- Provide unparalleled service and best value to their customers through a responsive and cost effective supply chain.
- Committed to provide quality products by strict adherence to international standards and best practices through technical collaboration with leading global companies in this business.
- Maximize shareholders value through sustained profitable growth.
- Aggressively focusing on increasing penetration in the piping systems market by exploring new channels.
- Company will continue to set new trends through innovative marketing and manufacturing.

The **company's vision** is to be the most valued company for all of its stakeholders, renowned for customer focus, innovation, quality, reliability and ethical practices.

At present PBP Ltd. does not have any secure method of saving company's customers data and information and most of the work are done on Microsoft excel sheets, word documents or on papers. Managing company's data administration is very important because it has a significant impact on efficiency and productivity in the whole company.

Recently, an IT study had been carried out by **Punjab Building Products Ltd**. As a result, it was decided that a complete development program has been started. The main reason to start this project was to advance the use of modern IT (Information Technology) in sales and marketing and warehousing. A large investment program is required to develop the information systems for sales and marketing. It will be executed by carrying out several different projects. One of these projects is the **Customer Information Management Project**, **CIM**. CIM is an ideal tool for businesses that need to securely save customer data and provide reports for marketing and sales purposes. The management of PBP Ltd. has agreed to launch the CIM project in January 2017. The objective of the project is to improve the quality of company's data administration, which will result in increased efficiency and productivity in the whole company.

1.2 Overview of Customer Information Management System

The main task and objective of the project is to design and build a prototype of the Customer information management system for **PBP** (Punjab Building Products Ltd).
The prototype will be used to test the functionality, usability and the features of the new system such as maintaining information about customers, products, suppliers and deliveries.

The main operator/ user of the system will be the staff members of the company.
The authorised users can log in to the system as an administrator which means that the customers don't have any access to the system because the system is purely designed for PBP Ltd. staff members. Customers can contact the company through phone call, fax or email.
Prototype of the customer information management system consists of five different user interfaces (UI) with different features which includes:

- Login Page
- Customer Information
- Product Information
- Supplier Information
- Delivery Information

As mentioned above, the prototype consist of five different UI (user interfaces) and it uses the common and central database in the company which simplifies information flow between the departments in the company so that the information can be saved and retrieved accordingly without any complications. The application will be built / made from "scratch" (beginning).One of the main reason of starting this project from scratch is to implement all the programming knowledge that I have received during my learning and study period. In this way one can implement own choice of design and codes and it gives great opportunities for more advanced features that can also be added to the application later on. Moreover, CIM system has never been used at PBP ltd. so the new web application should be easy to navigate and easy to understand for all the users. In this scenario one of the best available option is to start developing web application from scratch according to the requirements received from the client.

The user interface of login page is the first page the user will get when connected to **PBP** (Punjab Building Products Ltd) CIM system. The login page contains company's logo and the name on the top header of the page. The login details which consist of username and password is used to login into the system. After logging in as an administrator, the company staff member has rights to access the entire CIM system which consists of information about company's customers, products, suppliers and information on deliveries of the products.

When customer sends a purchase request for his requirement of the products to the company. The authorised PBP staff member contacts the customer and saves the information related to the customer information page and reviews the customer's demand

from the stock position both in his own branch as well as online from factory stock inventory. If stocks are available the authorised user of the system will make arrangements for the product's delivery to the customer and sends an invoice to the customer. But, if the stocks available are not enough comparing to customer's demand then the PBP staff member will send a purchase requisition to the supplier for the desired product and get the products delivered to the company's stock inventory. During this process all the information regarding customers, products, suppliers and deliveries will be saved into the database of CIM system.

1.2.1 User Interface Overview

An overview of the user interface of CIM System exists as follows:

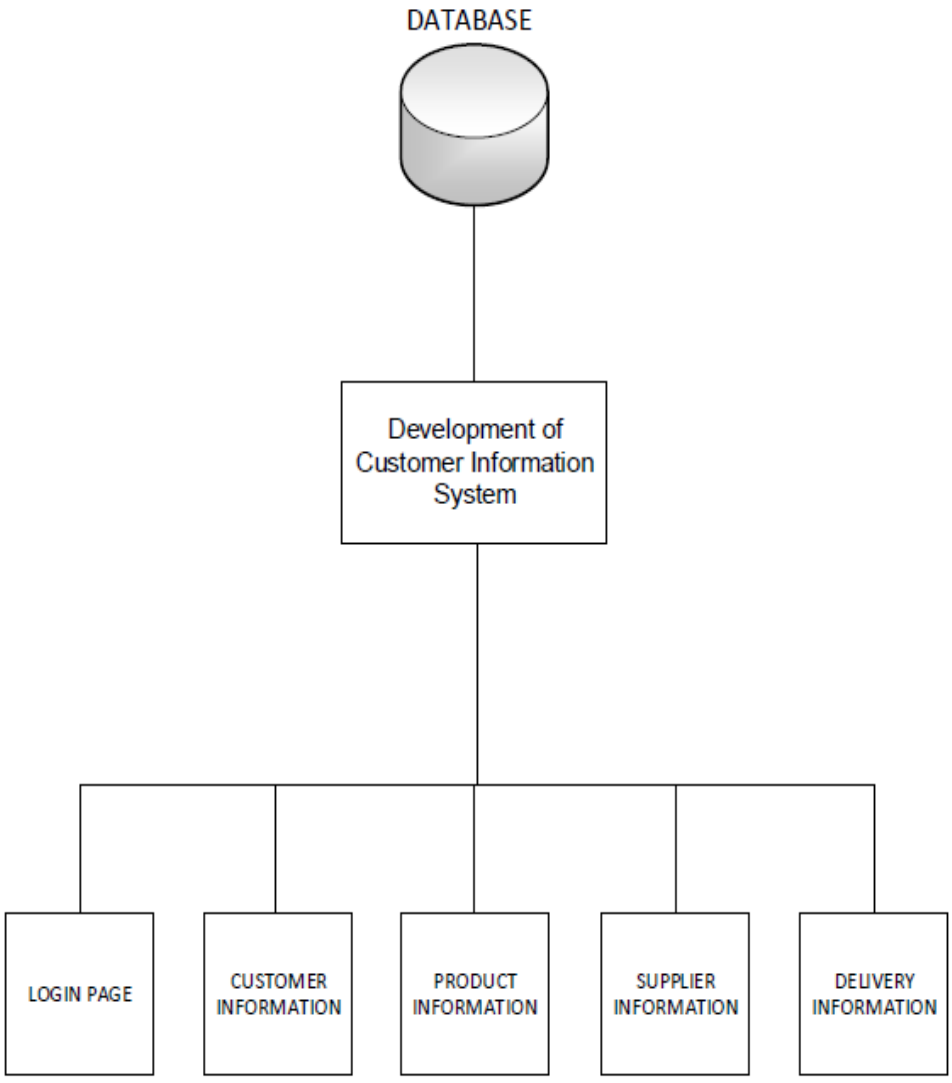

Figure 1. User Interface Overview

1.3 Objective and deliverables of the Project

The main task and objective of the project is to design and build a prototype of the customer information management system for **PBP** (Punjab Building Products Ltd). The prototype will be used to test the functionality, usability and the features of the new system such as maintaining customer's information, product's information, supplier's information, delivery information.

Project deliverables are:

- User interface and database of the system.
- System requirement documentation

1.4 Project Scope

Recently, PBP Ltd. was struggling to find a secure method of saving company's customers data and information because most of the work done on Microsoft excel sheets or word document, which was a big risk for the company information data security. The main reason to start this project was to advance the use of modern IT (Information Technology) in sales and marketing and warehousing. Customer information management system (**CIM**) is an ideal tool for businesses that need to securely save customer data and provide reports for marketing and sales purposes.

1.5 Out of Scope

The implementation of the final version of the product and deployment of the new system are not included in the project.

2 Stakeholders

The main stakeholders of the system are:

- Chairman PBP ltd.
- System administrator

The responsibilities of system administrator includes maintaining web server, web application, database, DBMS, security, installing updates of software versions and database backups.

2.1 The Client (Sponsor)

The main sponsor and client for the Customer information management system for PBP ltd. is the chairman of Punjab building products ltd. The client is the one who gives the final confirmation and then the product is developed.

2.2 Users of the System

The main users of the system will be the staff members of the company. Authorised users can log in to the system as an **administrator** which means that the customer's don't have any access to the system because the system is purely designed for PBP Ltd. staff members. Customers can contact the company through phone call, fax or email.

3 Terminology

3.1 Definitions

Following are the key terms definitions used by sponsors which are involved in the development of the new CIM system.

Term	Synonyms	Definition
Login page	Sign in page	Login options and login details include username and password.
Customer Information page	Consumer/ Buyer	Customer information which includes customer's name, address, contact Details, etc. are saved into the system.

Product Information page	Item for consumption	Information related to the products which includes reviewing customer's demand from the stock Position, checking availability of a product, updating company's stock inventory, etc. are saved into the system.
Supplier Information page	Vendor/ Dealer	Information related to the suppliers of the products are saved into the system.
Delivery Information page	Deliveries of products	Information related to the deliveries of the products are saved into the system.

Table 1: Definition of the Key Terms.

3.2 Conceptual Structure

The conceptual structure of the key terms is as follows:

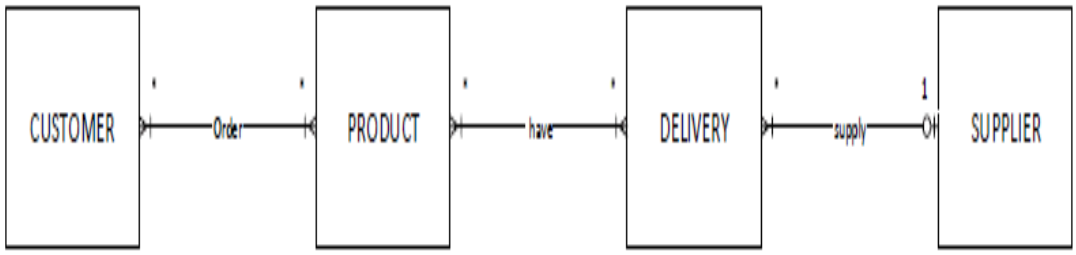

Figure 2. Conceptual Structure

4 System Architecture

The customer information management system for Punjab building products ltd has no subsystems and no dependencies of other system. All users will be using the system through a web browser.

The desired system architecture is shown in the figure below:

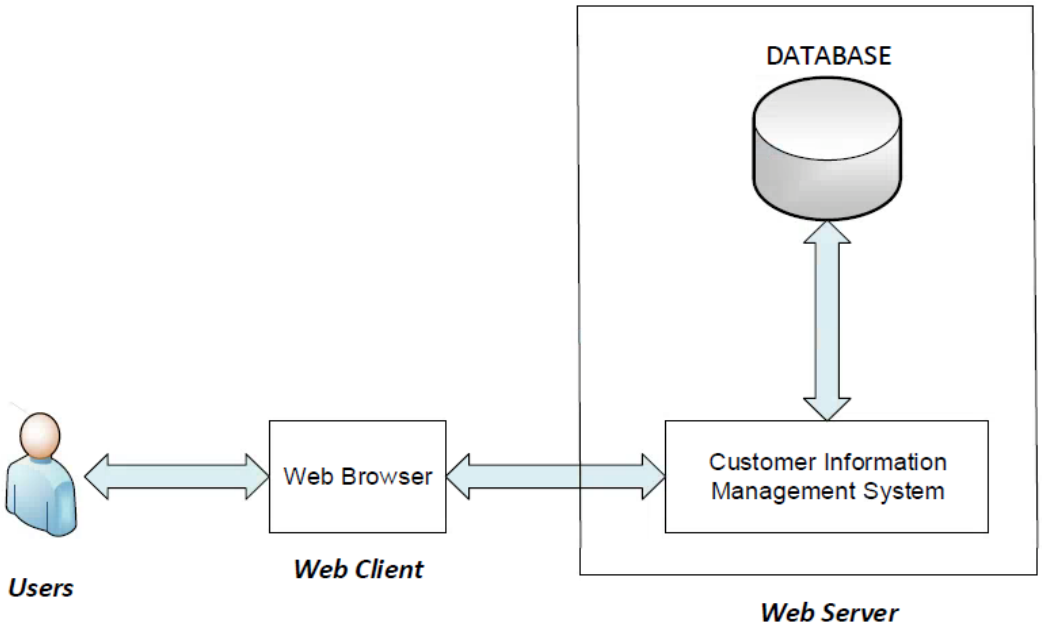

Figure 3. System Architecture

5 Technical Knowledge

Microsoft ASP.NET MVC framework is used in this project to develop the web application. The reason behind selecting Microsoft ASP.NET MVC framework is that as I already have decent knowledge and experience about MVC Frameworks so it will be more supportive and comfortable for me to put some more innovative features on the application in future implementation. Microsoft ASP.NET provides certified frameworks to create web applications. Basically all ASP.NET frameworks are created on the .NET Framework which share core functionality of .NET and ASP.NET. Frameworks provided by ASP.NET are established and advanced and a user can create great web applications with any of them. Individually each framework aims to develop great web applications but the target audience/developers of the application and the type of application developed can be different in these frameworks.

(MSDN Microsoft, 2015a.)

Following are the three frameworks provided by Microsoft ASP.NET for creating web applications:

- ASP.NET Web Forms
- ASP.NET MVC
- ASP.NET Web Pages

In this project, Microsoft ASP.NET MVC Framework 5.2 will be used to develop the web application. As mentioned above that there are many other frameworks in which the web application can be developed for example; ASP.NET Web Forms, ASP.NET Web Pages, PHP, etc. The reason behind selecting Microsoft ASP.NET MVC 5.2 Framework is that I already have decent knowledge and experience about MVC Frameworks so during the implementation phase it will be more supportive and comfortable for me to put some more innovative features on the application.

5.1 ASP.NET MVC Framework Overview

ASP.NET MVC framework is a lightweight framework that integrates with existing ASP.NET features and provides an alternative to the ASP.NET Web Forms for creating web applications.The architectural pattern of MVC (Model-View-Controller) framework is designed in such a way that it separates the application into three main components: **Model**, **View** and **Controller**.

By dividing the application into these 3 components, MVC makes it easier to manage complexity when building an application because it enables you to focus on one aspect of the implementation at a time. The work load can be divided and multiple teams can work on a web application because the code for the Business logic is separate from the code and mark-up for the presentation layer where developers can work on the business logic and designers can work on the mark-up and JavaScript. The pattern identifies where each kind of logic should be located in the application. The business logic belongs in the model. User interface logic belongs in the view and input logic belongs in the controller. (MSDN Microsoft, 2015b.)
Following figure (fig.4) gives an overview of ASP.NET MVC framework:

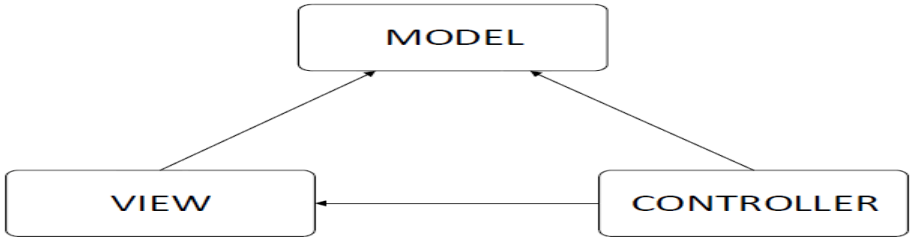

Figure 4. MVC Framework Overview

The MVC framework contains the following components:

Model: An MVC model contains all the application logic which includes business and data access logic .In specific, model objects implement the logic for the application's data domain. Model objects retrieve and store model state in a database Following are the two different methods to create model classes:

- Create EDM (Entity Data Model) from database.
- Create your own class.

Model class works as the link between the controller and the view. When a user activates a control it calls a method in the model to change its state and after changing the state it reports the view about the possible changes that occur. The model itself starts no activities but only accepts commands from the controller and process them. In order to work together with the application in easier way the same model can be used with different view/controller classes. (MSDN Microsoft, 2015b.)

View: View are the components that are responsible for displaying the application's user interface (UI) which includes web page structure, providing input controls and links which is created from the model data. In particular during the process when model informs that it has changed its state which activates the view which displays itself by calling methods in the model that returns all the information that the view should display. (MSDN Microsoft, 2015b.)

Controller: Controllers are the components that are responsible for handling user interaction, working with the model and select a view to extract which displays user interface. In MVC application the controller handles and responds to user input and interaction whereas the view only displays information. (MSDN Microsoft, 2015b.)

5.2 Advantages

Following are the advantages of ASP.NET MVC Framework:

- By dividing an application into the model, the view and the controller, it makes it easier to manage complexity when building an application.
- Developers who want full control over the performance of an application, MVC is an ideal framework for them because it does not use server based forms.
- To process web application requests through a single controller it uses a front controller pattern which enables you to design an application which supports rich routing infrastructure.
- Provides improved support for test-driven development (TDD).

- It works great for web designers who need full control over the application behavior and also works well for web applications that are supported by large teams of developers.
  (MSDN Microsoft, 2015b.)

5.3 Security

As hackers had learnt to exploit bad codes it has created new security risks and threats. Following are some best practices that should be considered to secure an ASP.NET MVC Website:

- **Securing content:** Securing content of a site is very crucial especially when the content which is sensitive enough that there might be a chance of interception of HTTP packets, in this situation encryption must be used which will help to secure content on the site and keep hackers away from HTTP packets which can be intercepted. Encryption can be done manually before sending as cookies of many browsers are also encrypted in the same way.
  Password storage is another area where data encryption is required. Password should never be stored in clear text but always store a hash of the password which will help to secure your users data if in case password database is compromised.

- **Authentication:** It is the process which identifies the user of the site through a login process. When user logged in to the system, security tokens sent between the server and browser and authentication depend on those security tokens. Several authentication methods available for MVC are as follows:

  - No Authentication
  - Individual User Accounts (ASP.NET identity)
  - Organizational Accounts (Windows Server AD or Azure AD)
  - Windows Authentication (Intranet)

- **Authorization:** It is the process of assigning user roles that will allow user to access certain features of the application. User role includes admin, user, etc.

- **Custom Filters:** Custom filters can be used for different security measures, for example; to make sure that the request came from your site. This security measure can be accomplished if custom filters will be used.
  (R, Cravens, 2009.)

5.4 ASP.NET MVC Framework Architecture

When user enters some URL in the browser to make the request for accessing any resource from the server, the request comes to controller first where routing engine decides the handling of request and controllers. If required, controllers will contact with model for data, where model operates on database and return data to controller in way of business objects. Controller is also responsible of selecting the appropriate view and transferring the data to the selected view, where data will be populated and at the end user receives view from the controller.

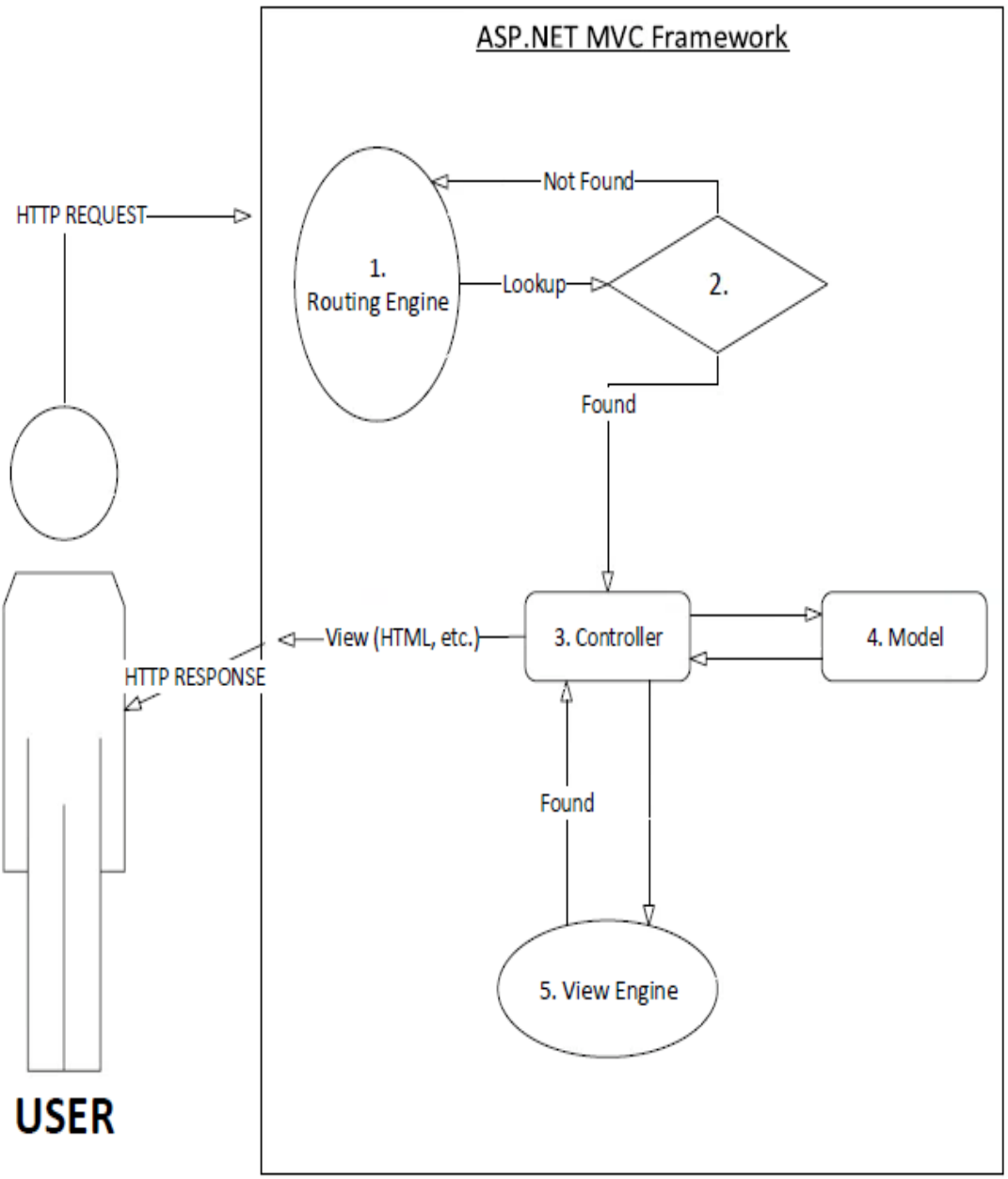

Figure 5. MVC Framework Architecture

5.5 Comparison between ASP.NET and PHP

Following are the factors on which Asp.net and PHP comparison will be prepared:

Factors	Asp.net	PHP
Rapid application development	Yes	No
Multiple programming languages	Yes	No
Object oriented language	Strong	Weak
Database support	Less	Many
Data controls	Yes	No
Platform	Windows only	Multiple platforms
Web servers	IIS only	Apache, IIS
Cost	High	Low
Support & Documentation	MSDN	Multiple sources
Exception Handling	Yes	Yes
Mobile applications support	High	Low
Web services support	Yes	Yes
Garbage collection	Best	Constructor & Destructor
Multi-threading	Yes	No

6 SRS (System Requirement Specification)

System requirement specification (SRS) is a complete explanation of the software system which is in under the development phase. To develop a software system, a developer should have the clear understanding of the requirement of the system. The SRS plays an important role in the development of any software system as it briefly describes the functions of the software system and performance potentials of the software system. The document describes the comprehensive vision and goals of the software system and functional and non-functional requirements of the software system which is essential for the success of any developed product or system.

The System requirement specification assembled from the PBP employee as the end user of the software system will be the staff members of Punjab building products ltd. Following are the information and requirements collected from the main sponsor and client for the customer information management system:

- The login page contains company's logo and name on the top header of the page.
- The login details which consist of username and password is used to login into the system.
- After logging in as an administrator the company staff member has rights to access the entire CIM system.

    - Browse Customer's Information.
    - Add new Customer Information.
    - Update existing Customer's Information.
    - Delete Customer's Information.

    - Browse Product's Information.
    - Add new Product Information.
    - Update existing Product's Information.
    - Delete Product's Information.

    - Browse Delivery Information.
    - Add new Delivery Information.
    - Update existing Delivery Information.
    - Delete Delivery Information.

    - Browse Supplier's Information.
    - Add new Supplier Information.
    - Update existing Supplier's Information.
    - Delete Supplier's Information.

6.1 Use cases for System Requirement

A methodology used in system analysis to identify, clarify and organize system requirements is known as use case methodology. A use case model is a graphical description of the interactions among the elements of a system or in other words a use case model describes the proposed functionality of the new system. Use case models works in UML (Unified Modeling Language) which is a standard notation for the modeling of objects and systems. (Rouse, 2007.)

Use case is a single unit of meaningful work and provides a good way of getting an overall picture of what is happening in the existing system or what is planned to happen in the new system. Use case models are very simple and it's an effective way of communicating with users and other stakeholders about the system that what is planned to do.

6.1.1 Use cases

Use cases represents sequence of actions carried out by the system. Each use case have a description which describes the functionality that will be built in the proposed system. A use case may 'include' another use case's functionality or 'extend' another use case with its own behavior. (Sparx Systems 2004.)

6.1.2 Actors

An actor is a user of the system. Actors represent the people or other machines that interact with the system being modelled to perform meaningful work. Use cases are usually related to actors and it uses a use case to perform some piece of work which is of value to the business. The overall role in the system and the scope of action of an actor is determined according to the set of use cases an actor has access to. (Sparx Systems 2004.)

7 Use Case Diagram

The main functions of the customer information management system for Punjab building products ltd. are shown as use cases in the use case diagram below:

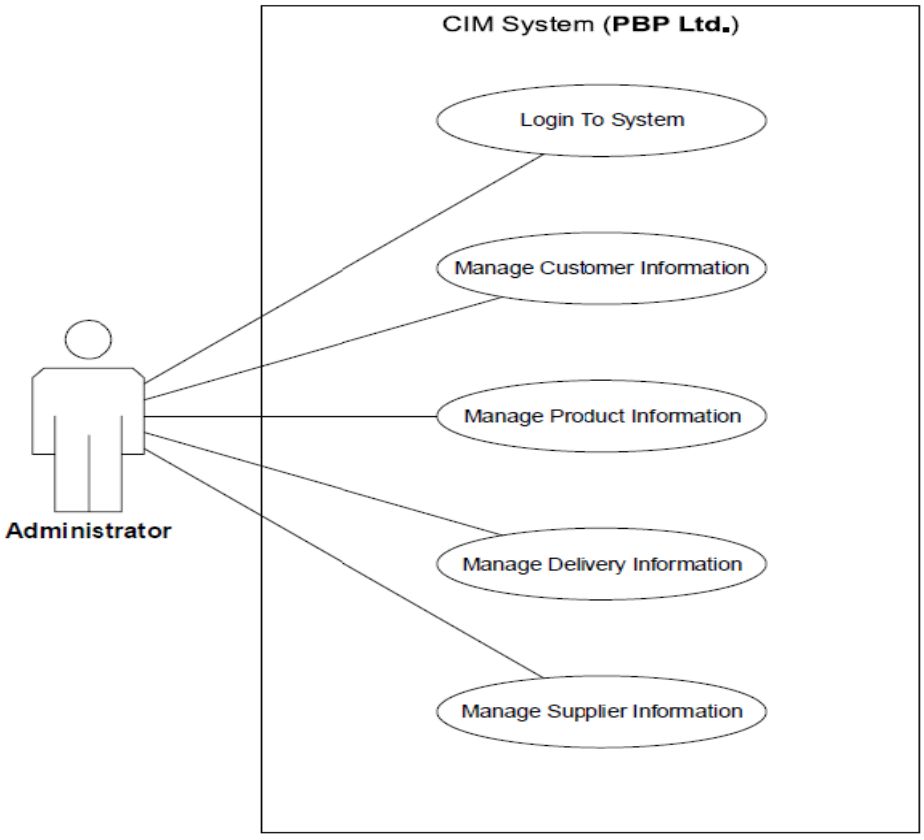

Figure 6. Use case Diagram

Actors

Administrator = PBP Ltd. staff member (board member, office assistant)

7.1 Services

Following services are provided by the system:

7.1.1 Login to System

The staff members of the company can log in to the system as an administrator. After Logging in as an administrator the company staff member has rights to access the entire CIM system which contains information about company's customers, products, suppliers and information on deliveries of the products.

7.1.2 Manage Customer Information

The administrator can browse customer information in the system. The administrator can also add new customer information to the system. The customer information can be updated later and customer information can be deleted from the system. Only authorised user should be allowed to manage customer information.

7.1.3 Manage Product Information

The administrator can browse product information in the system. The administrator can also add new product information to the system. The product information can be updated later and product information can be deleted from the system. Only authorised user should be allowed to manage product information.

7.1.4 Manage Delivery Information

The administrator can browse delivery information in the system. The administrator can also add new delivery information to the system. The delivery information can be updated later and delivery information can be deleted from the system. Only authorised user should be allowed to manage delivery information.

7.1.5 Manage Supplier Information

The administrator can browse supplier information in the system. The administrator can also add new supplier information to the system. The supplier information can be updated later and supplier information can be deleted from the system. Only authorised user should be allowed to manage supplier information.

8 Use-Case Dependency Diagram

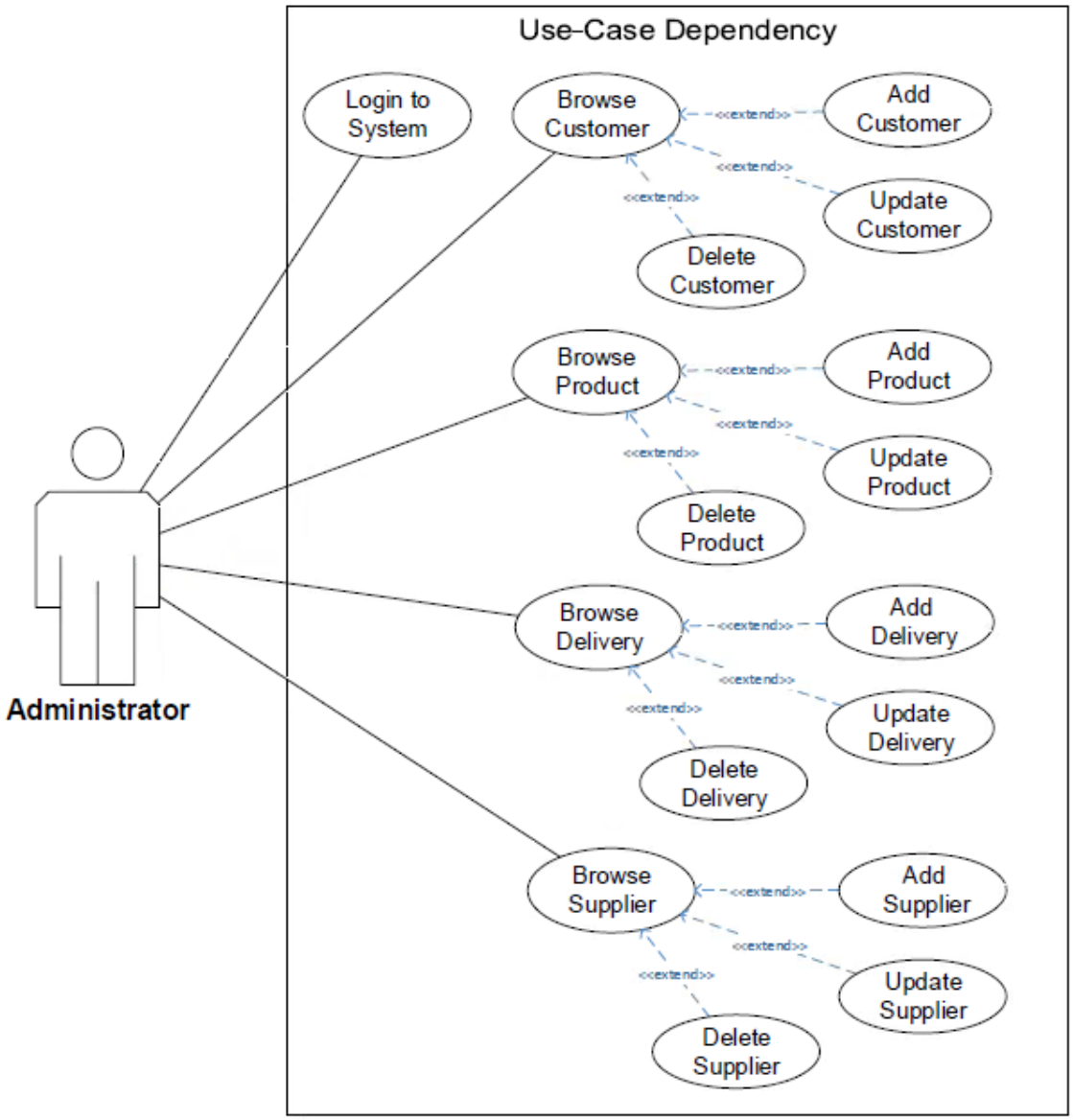

Figure 7. Use case dependency diagram

8.1 Use case Descriptions

8.1.1 Login to system

USE CASE NAME	Login to system
Includes / Extends	-
Overview	Logging into the system
Trigger	-
Actors	Administrator
Preconditions	The actor has Login IDs information but has not logged in so far.

Postconditions	The actor have successfully logged in as an authorised user of the system.
Basic Flow	
1	The actor directs to the login page where system displays the login form.
2	The actor enters the username and password on the login form and clicks the login button.
3	The system validates the actor login details.
4	A message is displayed to inform that the actor has successfully logged in.
Alternative flows	
The username/password not valid => system shows error message on (Step 3).	

8.1.2 Browse Customer

USE CASE NAME	Browse customer
Includes / Extends	Extends: add customer, update customer, delete customer
Overview	Viewing customer information
Trigger	-
Actors	Administrator
Preconditions	The user has logged in as an administrator.
Postconditions	The details of the selected customer are shown to the actor.
Basic Flow	
1	The actor opens the customer information viewing interface.
2	The system shows the list of customers (Customer ID, name)
3	The actor selects one of the customer from the customers list.
4	The system shows details of the selected customer.
Alternative flows	
The actor cannot find the customer from the list and exits. The actor decides to continue directly to the use case Add customer.	

8.1.3 Add Customer

USE CASE NAME	Add customer
Includes / Extends	-
Overview	Adding one new customer into the database.

Trigger	A new customer sends a request by email or phone call providing the information to be added in the system.
Actors	Administrator
Preconditions	The user has logged in as an administrator
Postconditions	The new customer is successfully saved into the database.
Basic Flow	
1	The actor requests the system to show empty customer Information entry form.
2	The system shows an empty form for entering the customer information.
3	The actor enters the requested data and notifies the system that all data has been entered.
4	The system saves the data in the database.
5	A message is displayed to inform the actor that the customer data entered was successfully saved.
Alternative flows	
The actor enters invalid data => system shows an error message (Step3). Mandatory input data item missing => system shows an error message (Step3). Data already exists in the database => system shows an error message (Step3).	

8.1.4 Update Customer

USE CASE NAME	Update customer
Includes / Extends	-
Overview	Edit customer information
Trigger	An existing customer sends a request with the new customer information to be updated in the system.
Actors	Administrator
Preconditions	The user has logged in as an administrator
Postconditions	The customer information is successfully updated in the database.
Basic Flow	
1	The actor opens the customer information viewing interface where the system shows the list of customers (Customer ID, name)
2	The actor selects the desired customer from the customers list whose information needs to be updated.

3	The system shows details of the selected customer.
4	The actor provides the system with updated customer Information.
5	The system saves the data in the database and a message is displayed to inform the actor that the customer data entered was successfully updated.
Alternative flows	
The actor enters invalid data => system shows an error message (Step4). Mandatory input data item missing => system shows an error message (Step4). Data already exists in the database => system shows an error message (Step4).	

8.1.5 Delete Customer

USE CASE NAME	Delete customer
Includes / Extends	-
Overview	Removing customer information from database.
Trigger	In-active customers deleted annually.
Actors	Administrator
Preconditions	The user has logged in as an administrator
Postconditions	Customer is successfully deleted from the database.
Basic Flow	
1	The actor opens the customer information viewing interface where the system shows the list of customers (Customer ID, name)
2	The actor selects the desired customer from the customers list whose information needs to be deleted.
3	The system shows details of the selected customer.
4	The actor deletes customer information from the database.
5	A message is displayed to inform the actor that the Customer is successfully deleted from the database.
Alternative flows	

8.1.6 Browse Product

USE CASE NAME	Browse product
Includes / Extends	Extends: add product, update product, delete product
Overview	Viewing product information
Trigger	-
Actors	Administrator
Preconditions	The user has logged in as an administrator.
Postconditions	The details of the selected product are shown to the actor.
Basic Flow	
1	The actor opens the product information viewing interface.
2	The system shows the list of products (Product ID, name)
3	The actor selects one of the product from the products list.
4	The system shows details of the selected product.
Alternative flows	
The actor cannot find the product from the list and exits. The actor decides to continue directly to the use case Add product.	

8.1.7 Add Product

USE CASE NAME	Add product
Includes / Extends	-
Overview	Adding one new product into the database.
Trigger	A new product is received from the suppliers and Information related to the product will be added in the system.
Actors	Administrator
Preconditions	The user has logged in as an administrator
Postconditions	A new product is successfully saved into the database.
Basic Flow	
1	The actor requests the system to show empty product Information entry form.
2	The system shows an empty form for entering the product information.
3	The actor enters the requested data and notifies the system that all data has been entered.
4	The system saves the data in the database.

5	A message is displayed to inform the actor that the product data entered was successfully saved.
Alternative flows	
The actor enters invalid data => system shows an error message (Step3). Mandatory input data item missing => system shows an error message (Step3). Data already exists in the database => system shows an error message (Step3).	

8.1.8 Update Product

USE CASE NAME	Update product
Includes / Extends	-
Overview	Edit product information
Trigger	-
Actors	Administrator
Preconditions	The user has logged in as an administrator
Postconditions	The product information is successfully updated in the database.
Basic Flow	
1	The actor opens the product information viewing interface where the system shows the list of products (Product ID, name)
2	The actor selects the desired product from the products list whose information needs to be updated.
3	The system shows details of the selected product.
4	The actor provides the system with updated product information.
5	The system saves the data in the database and a message is displayed to inform the actor that the product data entered was successfully updated.
Alternative flows	
The actor enters invalid data => system shows an error message (Step4). Mandatory input data item missing => system shows an error message (Step4). Data already exists in the database => system shows an error message (Step4).	

8.1.9 Delete Product

USE CASE NAME	Delete product
Includes / Extends	-
Overview	Removing product information from database.

Trigger	-
Actors	Administrator
Preconditions	The user has logged in as an administrator
Postconditions	A product is successfully deleted from the database.
Basic Flow	
1	The actor opens the product information viewing interface where the system shows the list of products (Product ID, name)
2	The actor selects the desired product from the products list whose information needs to be deleted.
3	The system shows details of the selected product.
4	The actor deletes product information from the database.
5	A message is displayed to inform the actor that the product is successfully deleted from the database.
Alternative flows	

8.1.10 Browse Delivery

USE CASE NAME	Browse delivery
Includes / Extends	Extends: add delivery, update delivery, delete delivery
Overview	Viewing delivery information
Trigger	-
Actors	Administrator
Preconditions	The user has logged in as an administrator.
Postconditions	The details of the selected delivery are shown to the actor.
Basic Flow	
1	The actor opens the delivery information viewing interface.
2	The system shows the list of deliveries (Delivery ID)
3	The actor selects one of the delivery from the deliveries list.
4	The system shows details of the selected delivery.
Alternative flows	
The actor cannot find the delivery from the list and exits. The actor decides to continue directly to the use case Add delivery.	

8.1.11 Add Delivery

USE CASE NAME	Add delivery
Includes / Extends	-
Overview	Adding one new delivery into the database.
Trigger	A product is ready for delivery so the information related to the delivery will be added in the system.
Actors	Administrator
Preconditions	The user has logged in as an administrator
Postconditions	Delivery information is successfully saved into the database.
Basic Flow	
1	The actor requests the system to show empty delivery Information entry form.
2	The system shows an empty form for entering the delivery information.
3	The actor enters the requested data and notifies the system that all data has been entered.
4	The system saves the data in the database.
5	A message is displayed to inform the actor that the delivery data entered was successfully saved.
Alternative flows	
The actor enters invalid data => system shows an error message (Step3). Mandatory input data item missing => system shows an error message (Step3). Data already exists in the database => system shows an error message (Step3).	

8.1.12 Update Delivery

USE CASE NAME	Update delivery
Includes / Extends	-
Overview	Edit delivery information
Trigger	-
Actors	Administrator
Preconditions	The user has logged in as an administrator
Postconditions	The delivery information is successfully updated in the database.
Basic Flow	
1	The actor opens the delivery information viewing interface where the system shows the list of deliveries (Delivery ID)

2	The actor selects the desired delivery from the deliveries list whose information needs to be updated.
3	The system shows details of the selected delivery.
4	The actor provides the system with updated delivery information.
5	The system saves the data in the database and a message is displayed to inform the actor that the delivery data entered was successfully updated.
Alternative flows	
The actor enters invalid data => system shows an error message (Step4). Mandatory input data item missing => system shows an error message (Step4). Data already exists in the database => system shows an error message (Step4).	

8.1.13 Delete Delivery

USE CASE NAME	Delete delivery
Includes / Extends	-
Overview	Removing delivery information from database.
Trigger	-.
Actors	Administrator
Preconditions	The user has logged in as an administrator
Postconditions	A delivery is successfully deleted from the database.
Basic Flow	
1	The actor opens the delivery information viewing interface where the system shows the list of deliveries (Delivery ID)
2	The actor selects the desired delivery from the deliveries list whose information needs to be deleted.
3	The system shows details of the selected delivery.
4	The actor deletes delivery information from the database.
5	A message is displayed to inform the actor that the delivery is successfully deleted from the database.
Alternative flows	

8.1.14 Browse Supplier

USE CASE NAME	Browse supplier
Includes / Extends	Extends: add supplier, update supplier, delete supplier
Overview	Viewing supplier information
Trigger	-
Actors	Administrator
Preconditions	The user has logged in as an administrator.
Postconditions	The details of the selected supplier are shown to the actor.
Basic Flow	
1	The actor opens the supplier information viewing interface.
2	The system shows the list of suppliers (Supplier ID, name)
3	The actor selects one of the supplier from the suppliers list.
4	The system shows details of the selected supplier.
Alternative flows	
The actor cannot find the supplier from the list and exits. The actor decides to continue directly to the use case Add supplier.	

8.1.15 Add Supplier

USE CASE NAME	Add supplier
Includes / Extends	-
Overview	Adding one new supplier into the database.
Trigger	A new supplier sends a request by email or phone call providing the information to be added in the system.
Actors	Administrator
Preconditions	The user has logged in as an administrator
Postconditions	The new supplier is successfully saved into the database.
Basic Flow	
1	The actor requests the system to show empty supplier information entry form.
2	The system shows an empty form for entering the supplier information.
3	The actor enters the requested data and notifies the system that all data has been entered.
4	The system saves the data in the database.

5	A message is displayed to inform the actor that the supplier data entered was successfully saved.
Alternative flows	
The actor enters invalid data => system shows an error message (Step3). Mandatory input data item missing => system shows an error message (Step3). Data already exists in the database => system shows an error message (Step3).	

8.1.16 Update Supplier

USE CASE NAME	Update supplier
Includes / Extends	-
Overview	Edit supplier information
Trigger	An existing supplier sends a request with the new supplier Information to be updated in the system.
Actors	Administrator
Preconditions	The user has logged in as an administrator
Postconditions	The supplier information is successfully updated in the database.
Basic Flow	
1	The actor opens the supplier Information viewing interface where the system shows the list of suppliers (Supplier ID, name)
2	The actor selects the desired supplier from the suppliers list whose information needs to be updated.
3	The system shows details of the selected supplier.
4	The actor provides the system with updated supplier information.
5	The system saves the data in the database and a message is displayed to inform the actor that the supplier data entered was successfully updated.
Alternative flows	
The actor enters invalid data => system shows an error message (Step4). Mandatory input data item missing => system shows an error message (Step4). Data already exists in the database => system shows an error message (Step4).	

8.1.17 Delete Supplier

USE CASE NAME	Delete supplier
Includes / Extends	-

Overview	Removing supplier information from database.
Trigger	-
Actors	Administrator
Preconditions	The user has logged in as an administrator
Postconditions	Supplier is successfully deleted from the database.
Basic Flow	
1	The actor opens the supplier Information viewing interface where the system shows the list of suppliers (Supplier ID, name)
2	The actor selects the desired supplier from the suppliers list whose information needs to be deleted.
3	The system shows details of the selected supplier.
4	The actor deletes supplier information from the database.
5	A message is displayed to inform the actor that the supplier is successfully deleted from the database.
Alternative flows	

9 Data Requirements

9.1 Entity – Relationship Diagram

"An entity-relationship diagram is a graphical representation of an information system that displays the relationship between entities such as people, objects, places, concepts or events within that system." In precise, an entity-relationship diagram is a data modelling technique which can help to define business processes and can be used as the foundation for a relational database. (Rouse, 2007.)

An entity-relationship diagram consist of **entities**, **attributes of the entities and relationship between entities** which are shown in an entity-relationship diagram below:

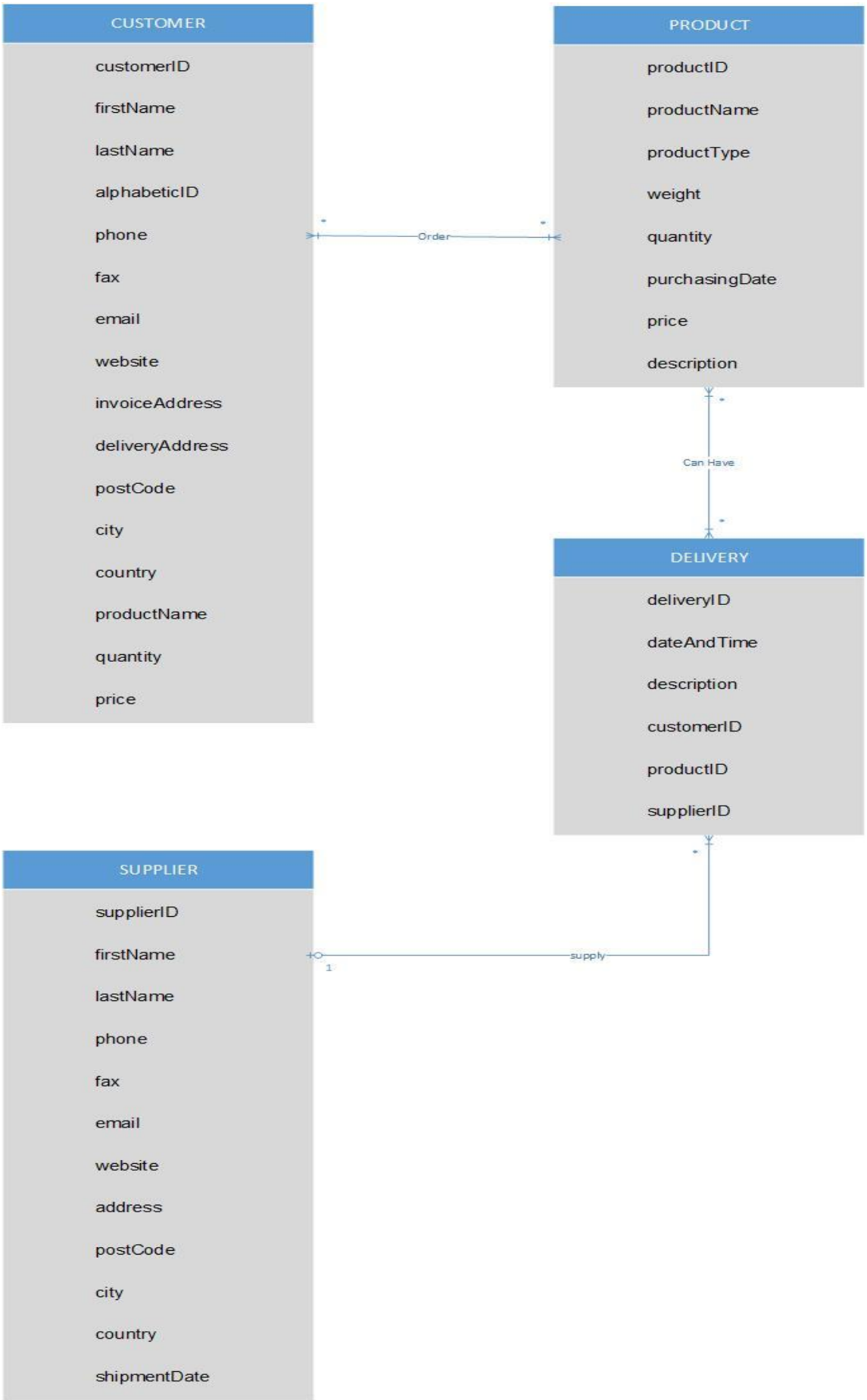

Figure 8. Entity - Relationship Diagram

9.2 Entity Type Descriptions

Entity type	Description	Synonyms	Occurrence
Customer	The customer orders product by phone call or email to the employee of the company. All information related to customers are saved into the database of the system.	Client / Buyer	Customer can be individual person or a business who orders and buys products from the company.
Product	Product is an Item that will be sold to the customer. All information related to products are saved into the database of the system.	Item / Goods	New products are added continuously throughout the year and old products are also deleted by the system administrator.
Delivery	Delivery contains amount of products that needs to be delivered. All information related to deliveries are saved into the database of the system.	Delivery	Each delivery can contain one or more than one products to be delivered.
Supplier	Supplier is the one who supply products to the company. All information related to suppliers are saved into the database of the system.	Vendor / Dealer	Supplier supply products that are needed in the Company's stock inventory.

9.3 Attribute Type Descriptions

9.3.1 Customer

Attribute Type	Description	Data type	Required	Special Domain
customerID	It is given by system, the system determines customer ID for new customers from continuous set of unique numbers. Identifies customer uniquely	Text	Yes	-
firstName	Customer First name	Text	Yes	-
lastName	Customer Last name	Text	Yes	-
alphabeticID	Alphabetical identification ID that has been formed from customer first and last name. Not formed Automatically but manually typed by user.	Text	Yes	-
phone	Customer phone number	Text	Yes	-
fax	Customer fax number	Text	-	-
email	Customer email address	Text	Yes	Valid email format
website	Customer website	Text	-	Valid website format
invoiceAddress	The address for invoicing of the customer.	Text	Yes	-
deliveryAddress	The address for delivery for the customer.	Text	Yes	-
postCode	Customer postcode	Text	Yes	-
city	Customer residential city	Text	Yes	-
country	Customer residential country	Text	Yes	-
productName	Product name	Text	Yes	-
quantity	Quantity of the ordered products	Integer	Yes	-

price	Price of the product	Integer	Yes	-

9.3.2 Product

Attribute Type	Description	Data type	Required	Special Domain
productID	A product ID is a number or string of alpha and numeric characters that uniquely identify a product.	Text	Yes	-
productName	Name of the product	Text	Yes	-
productType	Category of the product	Text	Yes	-
weight	Total weight of the product	Integer	Yes	-
quantity	Numbers of products purchased	Integer	Yes	-
purchasingDate	Date when the product is purchased	Date	Yes	Valid date format: yyyy-mm-dd
price	Price of the product	Integer	Yes	-
description	Detail description of the product	Text	Yes	-

9.3.3 Delivery

Attribute Type	Description	Data type	Required	Special Domain
deliveryID	A delivery ID is a number or string of alpha and numeric characters that uniquely identify a delivery.	Text	Yes	-
DateAndTime	Date and time of the delivery	DATETIME	Yes	Valid formats date: yyyy-mm-dd time: hh-mi-ss

description	Detail description of the delivery	Text	Yes	-
customerID	Uniquely Identifies customers	Text	Yes	-
productID	Uniquely Identifies products	Text	Yes	-
supplierID	Uniquely Identifies suppliers	Text	Yes	-

9.3.4 Supplier

Attribute Type	Description	Data type	Required	Special Domain
supplierID	It is given by system, the system determines Supplier ID for new suppliers from continuous set of unique numbers. Identifies supplier uniquely	Text	Yes	-
firstName	Supplier First name	Text	Yes	-
lastName	Supplier Last name	Text	Yes	-
phone	Supplier phone number	Text	Yes	-
fax	Supplier fax number	Text	-	-
email	Supplier email address	Text	Yes	Valid email format
website	Supplier website	Text	-	Valid website format
address	Business address of the supplier.	Text	Yes	-
postCode	Post code	Text	Yes	-
city	Supplier city of business	Text	Yes	-
country	Supplier country of business	Text	Yes	-
shipmentDate	Date of the shipment	Date	Yes	Valid date format: yyyy-mm-dd

9.4 Data CRUD Matrix

"A Data **CRUD** matrix **(Create, Read, Update,** and **Delete)** is a table that shows the links between processes and data. When a link exists, it shows the process performs create, read, update, or delete operation on the data." (Sybase, 2005.)

USE CASE	**DATA**			
	Customer	**Product**	**Delivery**	**Supplier**
Manage Customer Information	CRUD	-	-	-
Browse Customer	R	-	-	-
Add Customer	C	-	-	-
Update Customer	U	-	-	-
Delete Customer	D	-	-	-
Manage Product Information	-	CRUD	-	-
Browse Product	-	R	-	-
Add Product	-	C	-	-
Update Product	-	U	-	-
Delete Product	-	D	-	-
Manage Delivery Information	-	-	CRUD	-
Browse Delivery	-	-	R	-
Add Delivery	-	-	C	-
Update Delivery	-	-	U	-
Delete Delivery	-	-	D	-
Manage Supplier Information	-	-	-	CRUD
Browse Supplier	-	-	-	R
Add Supplier	-	-	-	C
Update Supplier	-	-	-	U
Delete Supplier	-	-	-	D

Note:

CRUD = Create, Read, Update, and **Delete**

10 Database Design

10.1 Mission Statement

In order for the company to keep track of customers, products, delivery, and supplier and generate reports for marketing a platform of logical data storage with similar entity needs to be defined. Such data storage needs to be credible, consistent, valid and flexible.

10.2 Mission Objectives

Managing database structure in different tier gives an opportunity for backup, recovery and security functions as well as providing flexibility for management. The access to the database is handled by the controller class database controller which handles the database queries and response. The main objectives are to store, update and view
Following entities:

- Customer
- Product
- Delivery
- Supplier

10.3 Logical Database Diagram

The logical design of the database is as follows:

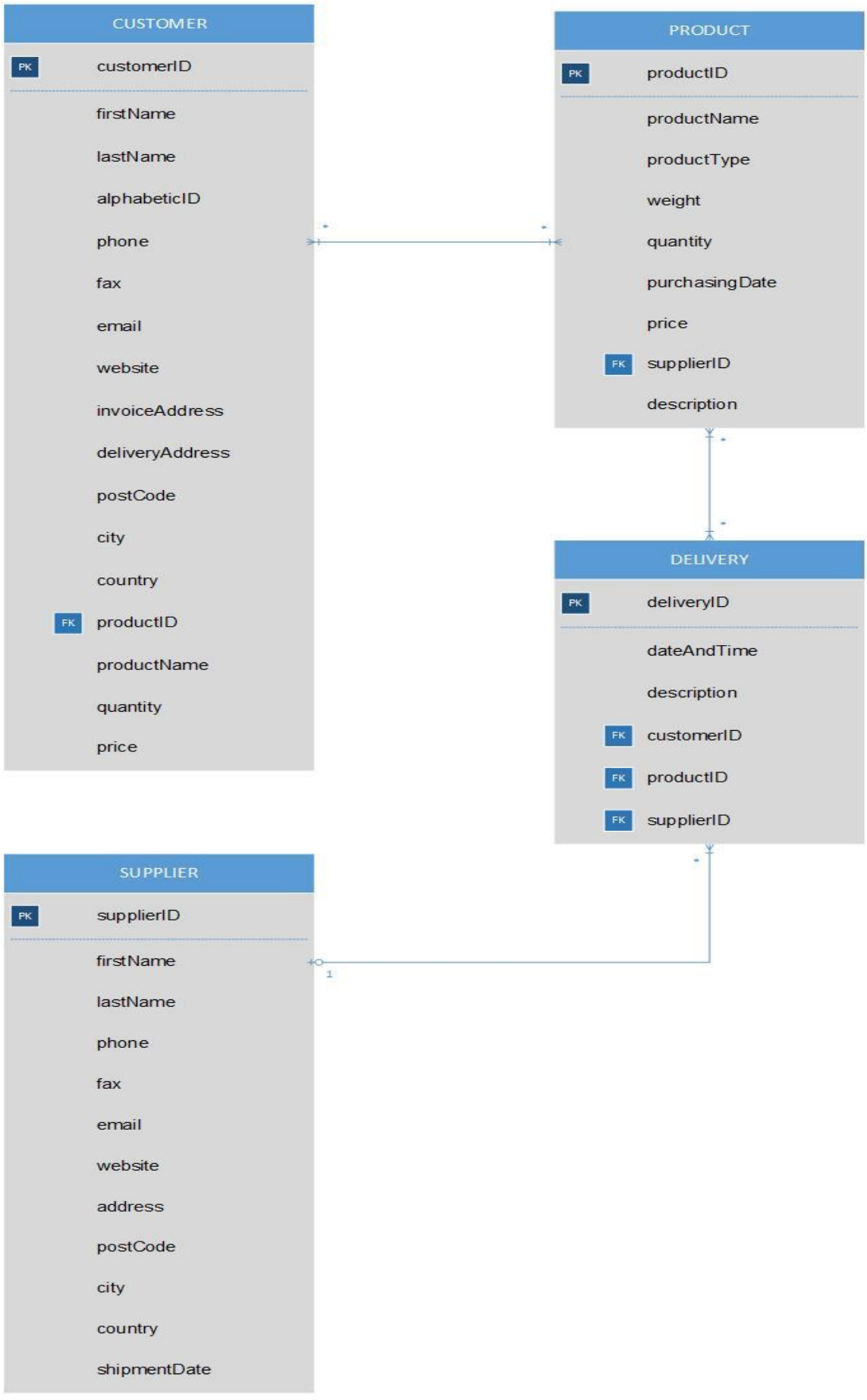

Figure 9. Logical Database Diagram

10.4 Data Dictionary

Data rules and conventions are mentioned below:

10.4.1 Customer

Name	Description	Data type & Length	Required	Special Domain	Constraints	Allow nulls
customerID	Identifies customer uniquely	int	Yes	101 - 9999	PK	N
firstName	Customer First name	Varchar (225)	Yes	-	-	N
lastName	Customer Last name	Varchar (225)	Yes	-	-	N
alphabeticID	It is formed from customer first and last name. Not formed automatically but manually typed by user.	Varchar (20)	Yes	-	-	N
phone	Customer phone number	Varchar (225)	Yes	-	-	N
fax	Customer fax number	Varchar (225)	-	-	-	Y
email	Customer email address	Varchar (225)	Yes	Valid email format	-	N
website	Customer website	Varchar (225)	-	Valid website format	-	
invoiceAddress	The address for	Varchar (225)	Yes	-	-	N

	Invoicing of the customer.					
deliveryAddress	The address for delivery for the customer.	Varchar (225)	Yes	-	-	N
postCode	Customer postcode	Varchar (10)	Yes	-	-	N
city	Customer residential city	Varchar (225)	Yes	-	-	N
country	Customer residential country	Varchar (225)	Yes	-	-	N
productID	Uniquely Identifies products	int	Yes	101 - 9999	FK	N
productName	Name of the product	Varchar (225)	Yes	-	-	N
quantity	Quantity of the ordered products	int	Yes	-	-	N
price	Price of the product	int	Yes	-	-	N

10.4.2 Product

Name	Description	Data type & Length	Required	Special Domain	Constraints	Allow nulls
productID	Uniquely Identifies products	int	Yes	101 - 9999	PK	N

productName	Name of the product	Varchar (225)	Yes	-	-	N
productType	Category of the product	Varchar (225)	Yes	-	-	N
weight	Total weight of the product	Varchar (20)	Yes	-	-	N
quantity	Quantity of the ordered products	int	Yes	-	-	N
purchasingDate	Date when the product is purchased	DATE	-	Valid date format: yyyy-mm-dd	-	N
price	Price of the product	int	Yes	-	-	N
supplierID	Uniquely Identifies suppliers	int	Yes	101 - 9999	FK	N
description	Detail description of the product	Varchar (225)	-	-	-	Y

10.4.3 Delivery

Name	Description	Data type & Length	Required	Special Domain	Constraints	Allow nulls
deliveryID	Uniquely Identifies delivery	int	Yes	101 - 9999	PK	N
dateAndTime	Date and Time of the delivery	DATETIME	Yes	Valid formats date: yyyy-mm-dd time: hh-mi-ss	-	N
description	Detail description	Varchar (225)	-	-	-	Y

	of the delivery					
customerID	Uniquely Identifies customers	int	Yes	101 - 9999	FK	N
productID	Uniquely Identifies products	int	Yes	101 - 9999	FK	N
supplierID	Uniquely Identifies suppliers	int	Yes	101 - 9999	FK	N

10.4.4 Supplier

Name	Description	Data type & Length	Required	Special Domain	Constraints	Allow nulls
supplierID	Identifies supplier uniquely	int	Yes	101 - 9999	PK	N
firstName	Supplier First name	Varchar (225)	Yes	-	-	N
lastName	Supplier Last name	Varchar (225)	Yes	-	-	N
phone	Supplier phone number	Varchar (225)	Yes	-	-	N
fax	Supplier fax number	Varchar (225)	-	-	-	Y
email	Supplier email address	Varchar (225)	Yes	Valid email format	-	N
website	Supplier website	Varchar (225)	-	Valid website format	.-	N
address	Business address of the supplier.	Varchar (225)	Yes	-	.	N

postCode	Post code	Varchar (10)	Yes	-	.	N
city	Supplier city of business	Varchar (225)	Yes	-	.	N
country	Supplier country of business	Varchar (225)	Yes	-	.	N
shipmentDate	Date of the shipment	DATE	Yes	Valid date format: yyyy-mm-dd	.	N

Note:
PK = Primary Key
FK = Foreign Key
Int = Integer
N = No
Y = Yes

11 User Interface Design

11.1 Interface Standard Design

11.1.1 Interface Metaphor

Figure 10. PBP Official Logo

11.1.2 Interface Objects

The main entities used in the development of the system are as follows:

- **Displayer:** PBP logo and label (CIM – Customer Information Management System), and text box (The text box can be used to display information of the customer, product, delivery, Supplier information retrieved from database)

- **Container:** The form for login page, search (homepage), customer information, product information, delivery information and supplier information.
- **Decorator:** Border to separate different section within the form.
- **Manipulator:** Scroll bar is used when the display content extend over the display area (generate automatically by the system)
- **Executor:** Buttons which execute command from user and display the result as instructed with involvement of different queries such as (login, search, browse, update, display, delete, submit, cancel, clear)
- **Navigator:** Buttons which simply navigate to different pages (customer information page, product information page, delivery information page, supplier information page).
- **Selector:** Check box
- **Editor:** Text box to accept input from the user.

11.1.3 Interface actions

Valid functions used in the use cases are as follows:

- **Login:** Login to the system.
- **Select:** Select customer information to be displayed.
- **Add:** Navigate to customer information page in order to create a new customer.
- **Display:** Retrieved data of customer being selected from the database and display this data in the text box of customer information page.
- **Update:** Display customer information in text box of customer page to allow user update and change the information.
- **Delete:** Select customer Information to be deleted.
- **Submit:** Get data from textbox and save the data into database.
- **Cancel:** Ignore all the action and return to the homepage.
- **Clear:** Clear all the textboxes to enter new data.
- **Back:** To return to the homepage.

11.2 Interface Template

This is the basic Interface layout for the Customer information management system:

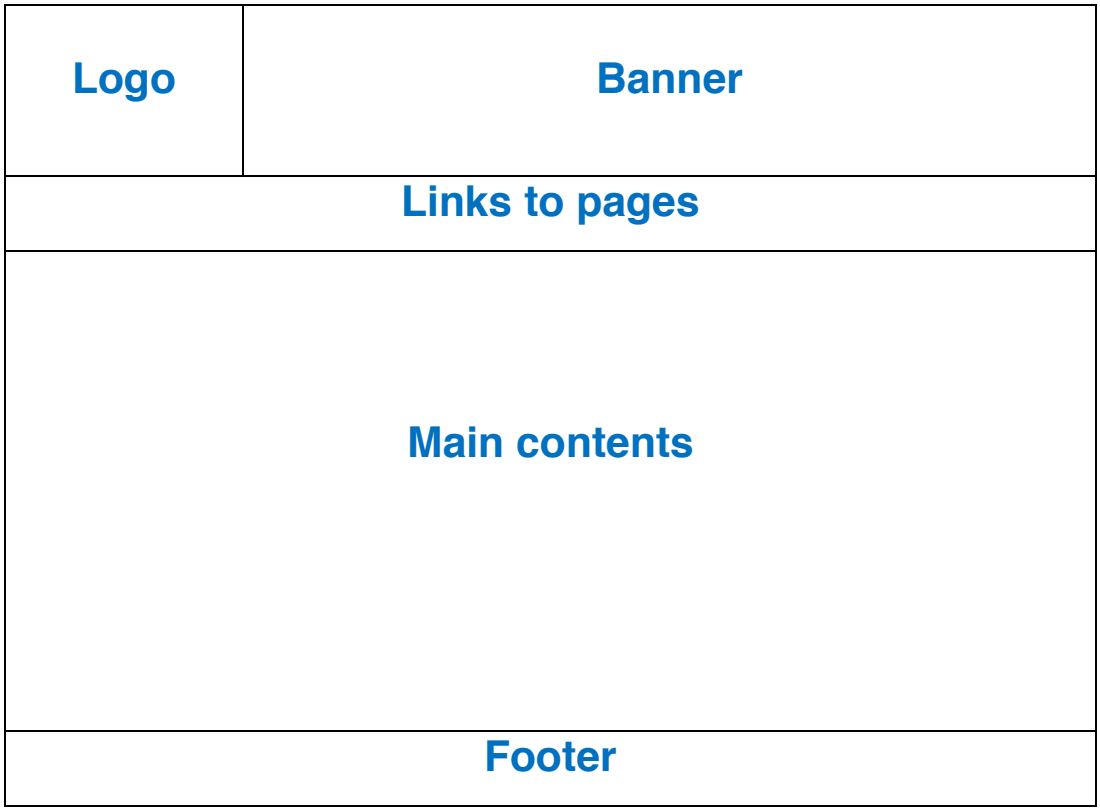

Figure 11. Interface Template

11.2.1 Principle of Navigation

In navigation menu there will be one main navigation title called home page which can be used for returning to main page.

11.2.2 The Messages (error, warning, confirmation)

- For inserting wrong type of data, error message will be displayed as a text beside the text field.
- For successful save, confirmation message will be displayed in a message box.
- **Error reporting:** red font colour with "*" mark next to the cause of error.
- **Errors** are managed by the error pop up window.

11.2.3 Colors and Fonts

Interface color palette used in the development of system are as follows:

Figure 12. Hexa Decimal Colors Palette

- **Font type:** Open-sans, Arial, Consolas, Happy-monkey, Monaco.
- **Font sizes**: Variable 10-25

11.3 Input Design

The user interface inputs the data in the following ways:

11.3.1 Types of inputs

Types of inputs used in the development of system are as follows:

- **Search box:** Text Box
- **Search criteria:** Button Box.
- **Data Input:** Text Box

11.3.2 Validation

The system validates the data in the following ways:

- **Update Button:** Data entered must be valid in order to update and save the new information.
- **Search Box:** Data typed must be coinciding with the search criteria requirements.

11.4 Output Design

The user interface displays the data in the following ways:

- Check box
- Select box
- Drop down list
- Text fields
- Search interface:
    - Search box
    - Button search

12 User Interface Design Prototype

12.1 Login Page

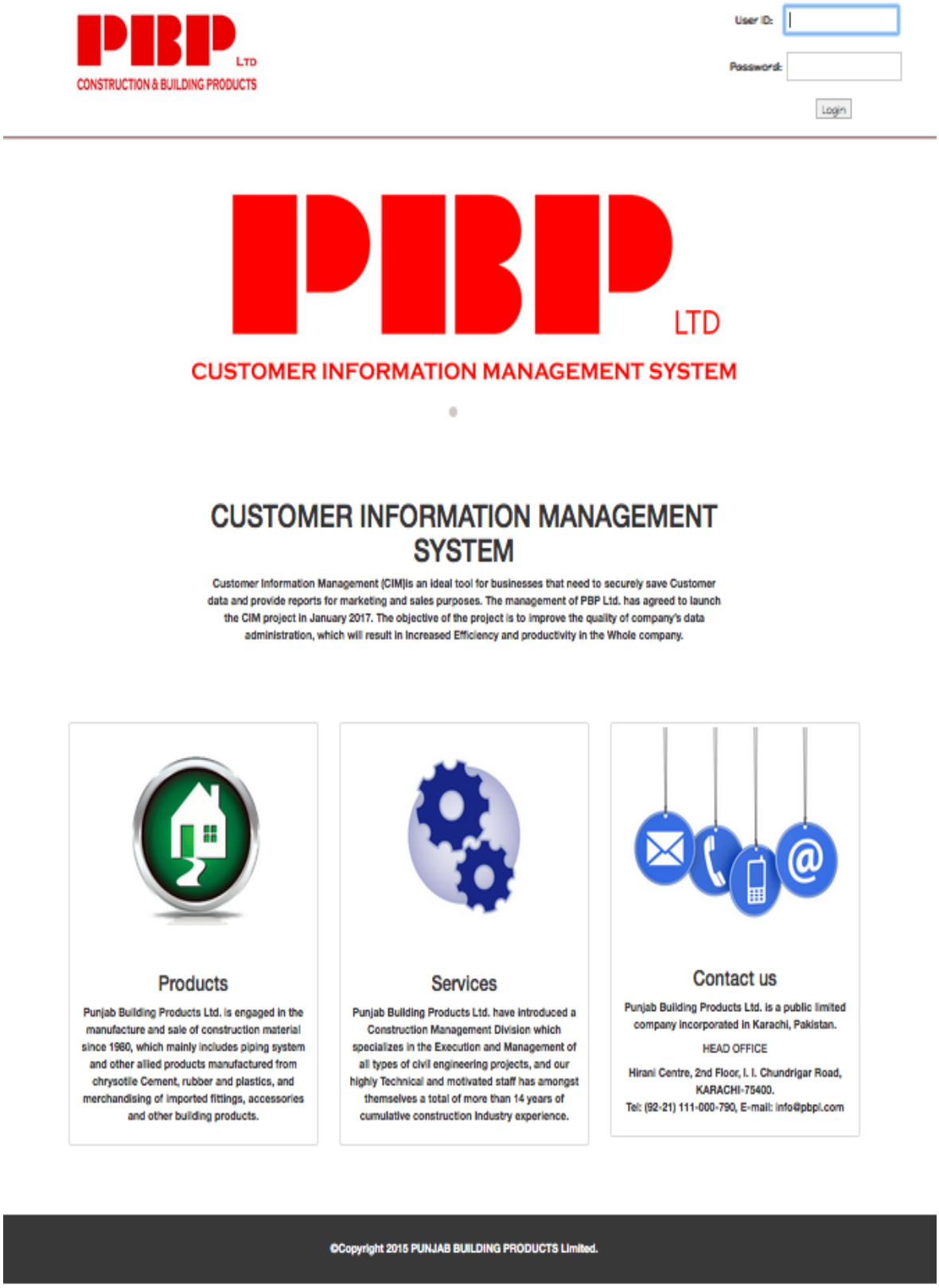

Figure 13. (UI) – Login Page

12.2 Customer Information Page

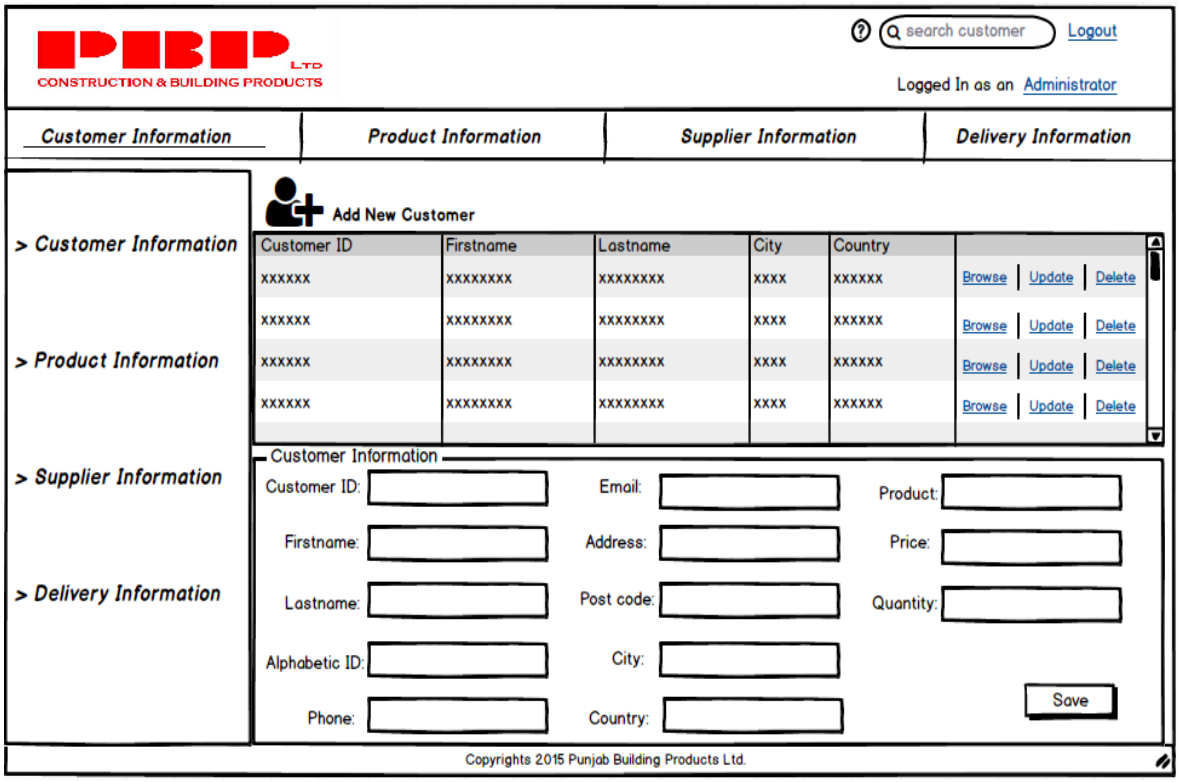

Figure 14. (UI) – Customer Information Page

12.3 Product Information Page

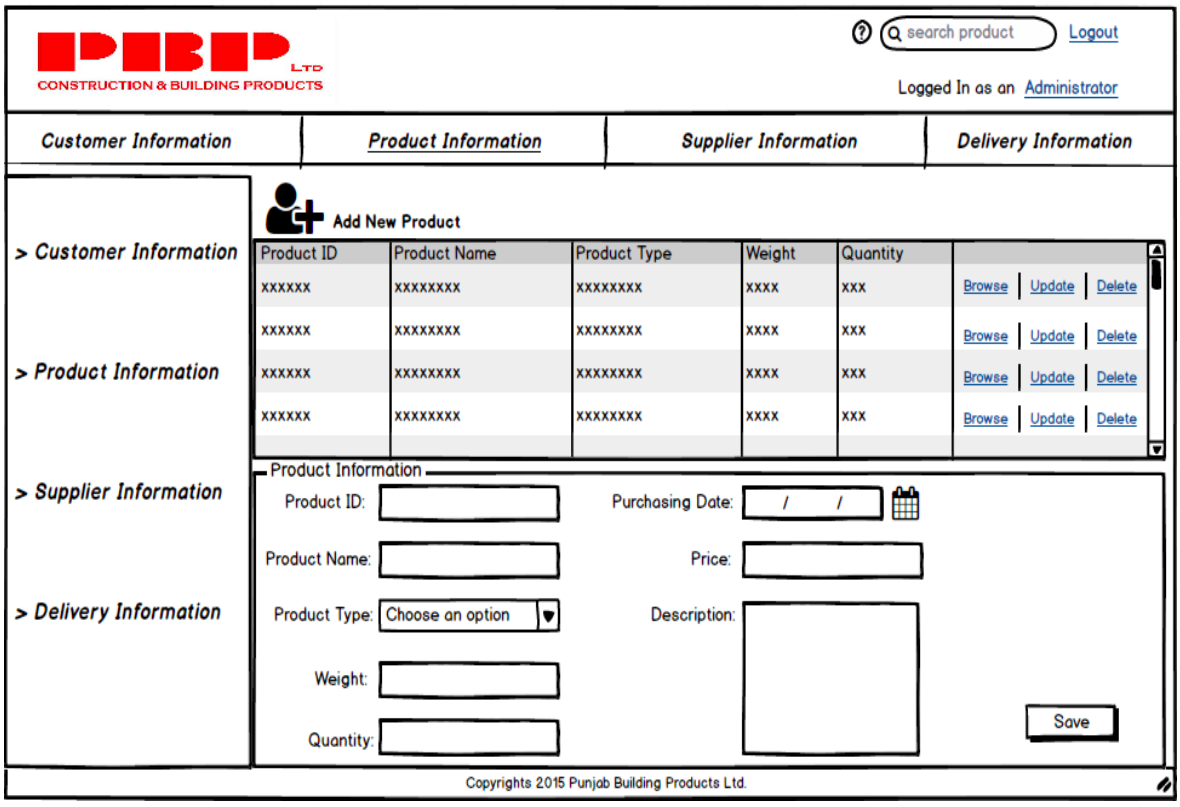

Figure 15. (UI) – Product Information Page

12.4 Supplier Information Page

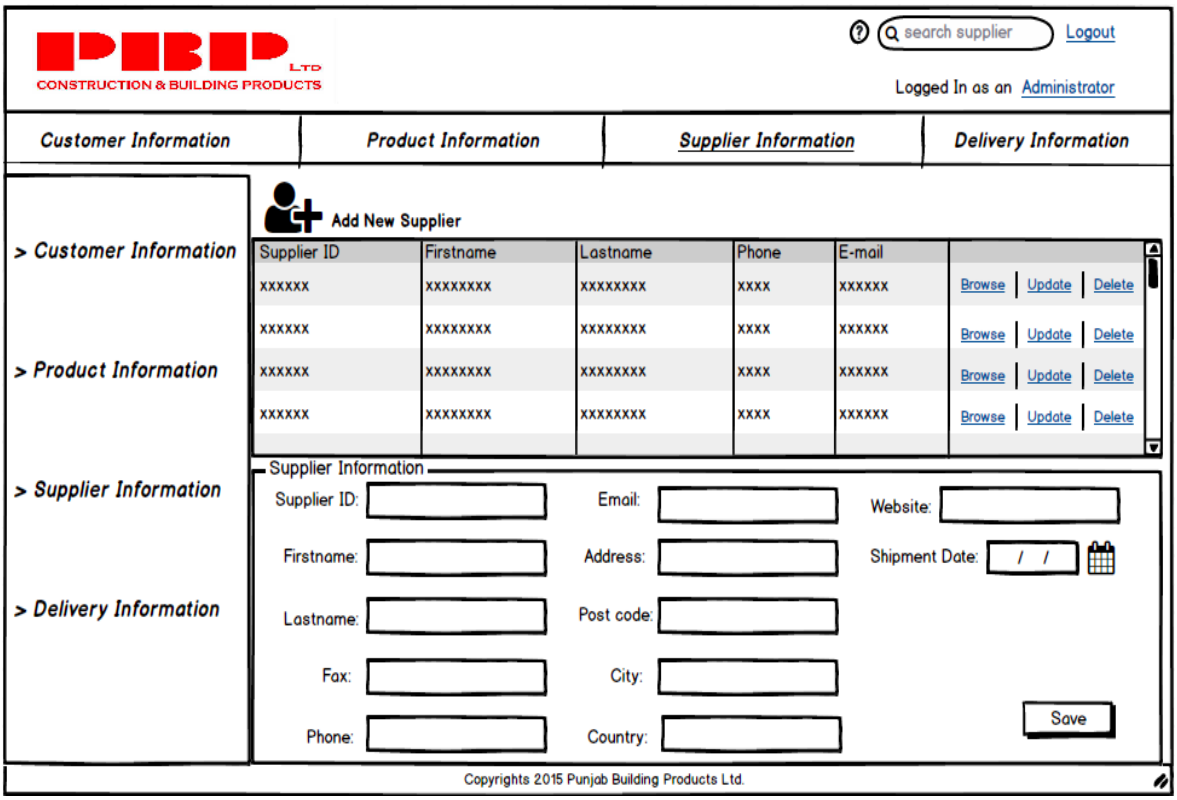

Figure 16. (UI) – Supplier Information Page

12.5 Delivery Information Page

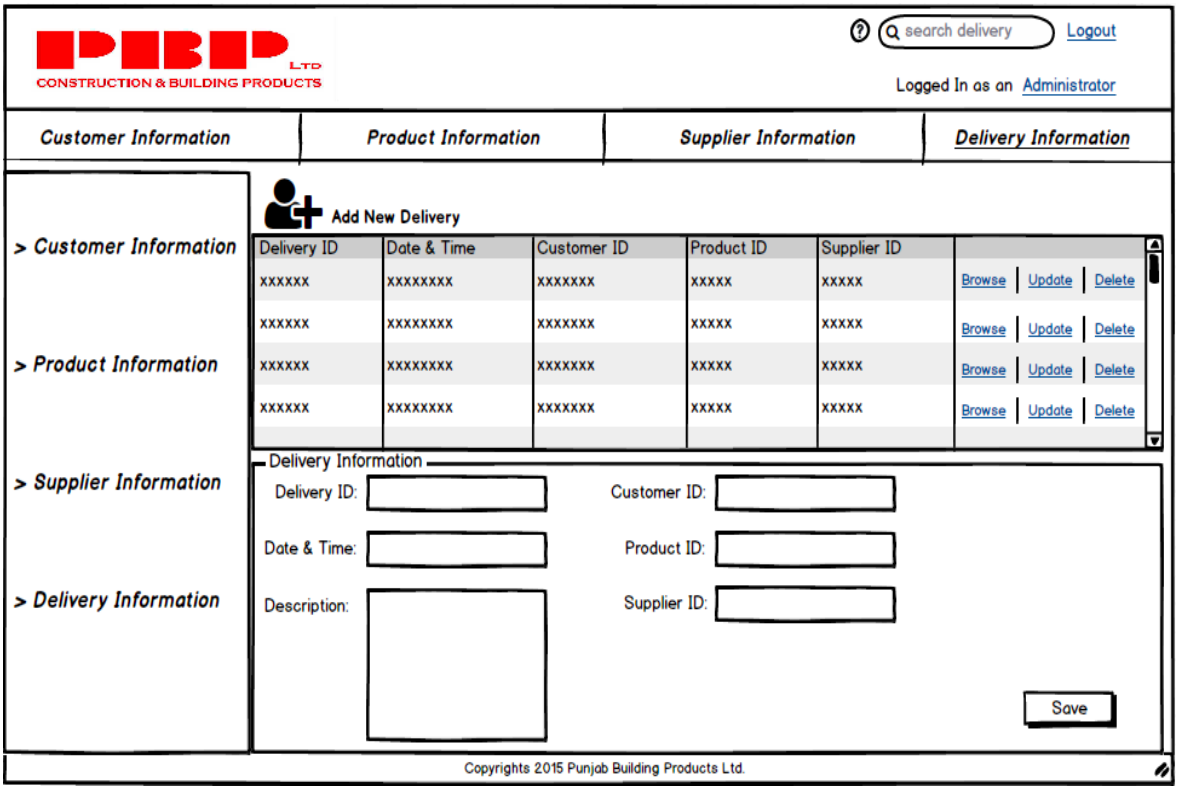

Figure 17. (UI) – Delivery Information Page

13 Usability Requirements

13.1 Usability Development Process

The usability standards presented here are the developer guidelines which are strictly followed during design and development process. The purpose of this process is to maximize work efficiency, reduce the amount of time needed for the learning curve and decrease the number of errors made by the users, thus diminishing any frustration and providing maximum user satisfaction. The main points of software usability standards are:

- Color scheme
- Layout
- Font and text standards
- Error Prevention
- Recovery

13.1.1 Color Scheme

Color scheme is consistent throughout the application. Colors used are soft and soothing, resulting in less eye strain. Color is used to make things stand out, while keeping the look aesthetically pleasing. Background is kept light-colored, without excessive brightness. Dark background color is not preferable, due to its inactive “dead” look. Borders are kept thin and are colored with a darker version of the background color, or shades of gray.
Any colors that are of high contrast are not usable, due to increased eye strain they cause thus will not be used throughout this software piece.

13.1.2 Layout

Effective layout is crucial in helping users find what they are looking for quickly, as well as making the appearance visually appealing. It describes the entire visual scheme of the web form, meaning it indicates where users need to look first. Spacing and size are very important parts of the layout. A correct way how to arrange button, text-box and list will avoid confusion and frustration.
An effective layout should include attributes described below:

- **Consistency:** Similar pages use a similar layout, so users always feel oriented.
- **Efficiency:** UI elements that important points need to be largely emphasized on and elements that are less important should be kept smaller.
- **Resizability:** Application should allow users to resize its‟ windows without losing the efficiency of the interface

13.1.3 Font Standard

Texts are kept clearly visible, of suitable size and color. The interface supports bold, italic and underlined text to emphasize a point. Textual styles are rarely used, due to their high attention-attraction factor.

Links are kept in Windows standard color scheme – e.g. links that were not visited are blue, active links are marked with red color and visited links are purple-colored.

Text color is used to pass subliminal messages, such as:

- **Red**: Hostility and disagreement. Used for error messages and important warnings.
- **Green**: Openness and sincerity. Text of green color is used for confirmation messages.
- **Gray**: Passiveness and inactivity. Used for tips and quick help pop-ups.

However, the bright text colors are not abused and/or misused, e.g. for exaggerated Messages.

13.1.4 Error Prevention

During Display all the information will be shown in inactive (non-editable) fields in order to avoid any mistake made by the user. Forcing the user to get used to with the inactive test boxes while displaying.

13.1.5 Recovery

For any error the user will get, messages giving sufficient information about the error and how to recover with a possible solution.

It allows the users to figure out the problems and quickly recover from them by the use of functions and message boxes.

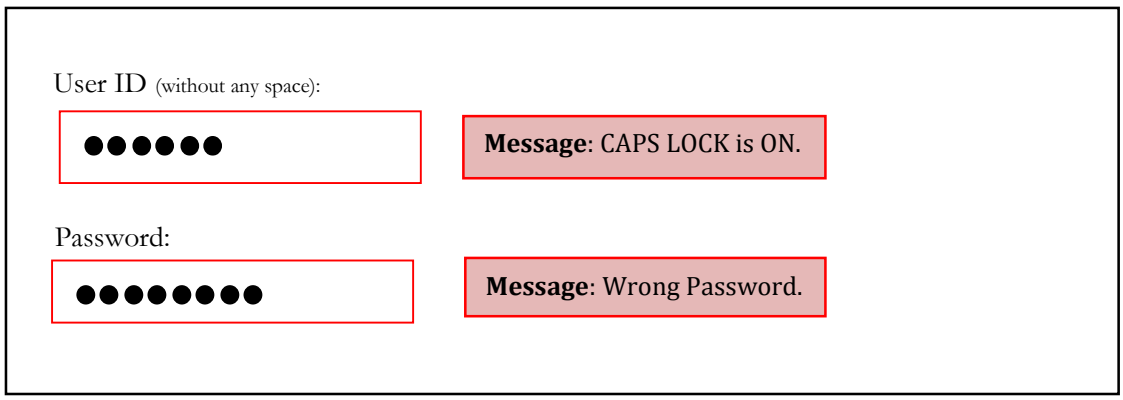

Figure 18. Recovery Information

14 Conclusion

Thesis is based on developing a Customer Information Management System for (PBP) Punjab building products ltd. The PBP ltd. is a public limited company and it started its activities in 1980 and involved in the manufacturing of building products. Since, C.I.M System is not used at PBP ltd., I have managed to give a complete computerized I.T support management system wherein all the functions and activities of information-storages, customer Data, information of all products/services, customer needs, suppliers information, stock-inventory, order-booking, processing of orders, deliveries to customers etc., is controlled, monitored and readily available to all concerned in the company in just one click.

I have used Microsoft ASP.NET MVC framework to develop the web application. The reason behind selecting Microsoft ASP.NET MVC framework is that as I already have decent knowledge and experience about MVC Frameworks so it will be more supportive and comfortable for me to put some more innovative features on the application in future implementation. The main users of this CIM System would be the staff members or administrators of the PBP ltd. To facilitate this, a prototype of the CIM system having five different UI (user- interfaces) are used whereby the information flow from the database is simplified.

To login, the first page has specific requirements to be entered like username and password after which staff member can access the entire CIM system. So in this way an administrator have access to login page, customer information page, product Information page, supplier information page and delivery information page. All users will be using this system through a web browser. The company staff member can always browse information, add any new information, update or delete any information relating to customers, products, deliveries and suppliers, but with a precaution that only those authorized persons / administrators should perform such changes in the CIM System via the use case dependency process.

Further, one cannot overlook the importance of Entity-Relationship diagram which is basically a data modelling technique which helps to define business processes and can be used as the foundation for a relational database. An entity-relationship consists of entities, attributes of the entities and relationship between entities. Further, we have also discussed Data CRUD Matrix (Create, Read, Update and Delete) showing the links between processes and data. When a link exists, it shows the process performs a CRUD operation on the data. In order for the company to keep track of customers, products, delivery, and supplier and generate reports for marketing, a platform of logical data storage with similar

entity needs to be defined. Such data storage needs to be credible, consistent, valid and flexible. Managing database structure in different tier gives an opportunity for backup, recovery and security functions as well as providing flexibility for management. The main objectives are to store, update and view following entities namely customer, product, delivery and suppliers.

The customer information management system can help you make better decisions by delivering all the information you need and by modeling the results of your decisions. Hence, itis very crucial to put the right information in front of right people at the right time so that they are enabled to make the right decisions. Also, it is imperative upon the decision makers to trust the available information. Since these information's are corporate assets, it has to be made secure at all cost.

14.1 Future Plans

After submitting the final version of thesis, I will continue to work on the system implementation and will finalize the application for company use. Minor changes may occur during the implementation phase so the requirements documents will be updated accordingly and the first version of the web application will be released in January 2016. Moreover, the final version of the web application is aimed to be released in January 2017 with some more advanced features.

15 References

Hexa Decimal Colors Palette, *Color Information*. URL: http://www.colorhexa.com
Accessed: 21 October 2015.

MSDN Microsoft 2015a, *ASP.NET Overview*. URL: https://msdn.microsoft.com/en-us/library/4w3ex9c2.aspx#wf_mvc_wp. Accessed 29 October 2015.

MSDN Microsoft 2015b, *ASP.NET MVC Framework*. URL: https://msdn.microsoft.com/en-us/library/dd381412%28v=vs.108%29.aspx. Accessed 29 October 2015.

R, Cravens 2009, *Security, Web*. URL: http://bobcravens.com/2009/08/6-security-steps-for-your-asp-net-mvc-web-site/. Accessed: 08 November 2015.

Rouse, M. Apr 2007, *Use Case Diagram*. URL: http://whatis.techtarget.com/definition/use-case-diagram. Accessed: 29 Sep 2015.

Sybase Inc. 2005, *Chapter 6 Building an Analysis Business Process Model,*
Defining the CRUD Matrix in an Analysis BPM.
URL:http://infocenter.sybase.com/archive/index.jsp?topic=/com.sybase.stf.powerdesigner.docs_12.0.0/html/bpug/bpugp124.htm. Accessed: 3 October 2015.

Sparx Systems 2004, *Use case Model*.
URL:http://www.sparxsystems.com.au/downloads/whitepapers/The_Use_Case_Model.pdf
Accessed: 27 Sep 2015.

Silpio, Kari 2015, *System Requirements and Software Requirements.*
Accessed: 5 October 2015.

W3school 2015, *HTML tutorial*: URL: http://www.w3schools.com/html/.
Accessed: 15 October 2015